IN THE SPECIFICATION:

Please delete the first sentence on page 1 and insert the following:

--RELATED APPLICATIONS

This application is a continuation of U.S. Patent Application Serial No. 08/708,945, filed September 6, 1996, now _____, which is a divisional of U.S. Patent Application Serial No. 08/410,357, filed March 24, 1995, now abandoned.--

IN THE CLAIMS:

Please cancel claims 12-16 and 23-43 without prejudice.

Please add new claims 44-60 as follows:

1   44. (New) A computer system that resolves name collisions by providing
2 type support for multiple type definitions, comprising:
3   an interface repository including:
4   a repository naming context; and,
5   a prefix naming context subordinate to the repository naming
6   context, the prefix naming context serving as a root naming
7   context for at least one interface definition language
8   declaration, the prefix naming context being adapted to re-
9   solve names subordinate to the repository naming context.

1   45. (New) The system of claim 44 wherein the prefix naming context
2 further includes:
3   at least one naming context defined by an interface definition object and
4   subordinate to the prefix naming context.

46. (New) The computer system of claim 45 wherein at least one interface definition object has a fully scoped object name including a prefix name of the prefix naming context to which the interface definition object is subordinated.

47. (New) The computer system of claim 44 wherein the prefix naming context is immediately subordinate to the repository naming context.

48. (New) The computer system of claim 44 wherein the prefix naming context further includes:

at least one leaf node defined by an interface definition object.

49. (New) The computer system of claim 44, wherein the prefix naming context is defined by a prefix object.

50. (New) The computer system of claim 44, further comprising:

an interface repository loader that accepts as input parameters a specified interface definition language file containing at least one interface definition language declaration, and a specified prefix name, and installs the at least one interface definition language declaration in a prefix naming context having the prefix naming context in the interface repository.

51. (New) A computer system that resolves name collisions by providing type support for multiple type definitions, comprising:

an interface repository including:

a repository naming context; and

5   a prefix naming context subordinate to the repository naming

6   context, the prefix naming context serving as a root naming

7   context for at least one interface definition language

8   declaration, the prefix naming context being adapted to re-

9   solve names subordinate to the repository naming context;

10   and

11   an interface repository loader that accepts as input parameters a specified

12   interface definition language file containing at least one interface

13   definition language declaration, and a specified prefix name, and

14   installs the at least one interface definition language declaration in

15   a prefix naming context having the prefix naming context in the

16   interface repository, and wherein the interface repository loader

17   creates a data file identified as related to the specified interface

18   definition language file, and containing an identification of the

19   specified prefix naming context.

1   52. (New) A computer system that resolves name collisions by providing

2 type support for multiple type definitions, comprising:

3   an interface repository including:

4   a repository naming context; and

5   a prefix naming context subordinate to the repository naming

6   context, the prefix naming context serving as a root naming

7   context for at least one interface definition language

8   declaration, the prefix naming context being adapted to re-

9   solve names subordinate to the repository naming context;

10   and

11   an interface repository loader that accepts as input parameters a specified

12   interface definition language file containing at least one interface

13   definition language declaration, and a specified prefix name, and

14   installs the at least one interface definition language declaration in

15   a prefix naming context having the prefix naming context in the

16   interface repository, and wherein the interface repository loader

17   creates the specified prefix naming context in the interface

18   repository if the specified prefix naming context does not exist

19   therein.

1   53. (New) The computer system of claim 52, further comprising:

2   a memory device that stores the interface repository; and

3   a processing unit that executes operations of the interface repository

4   loader.

1   54. (New) A computer system that resolves name collisions by providing

2   type support for multiple type definitions, comprising:

3   an interface repository including:

4   a repository naming context; and

5   a prefix naming context subordinate to the repository naming

6   context, the prefix naming context serving as a root naming

7   context for at least one interface definition language

8   declaration, the prefix naming context being adapted to re-

9   solve names subordinate to the repository naming context;

10   an interface repository loader that accepts as input parameters a specified

11   interface definition language file containing at least one interface

12   definition language declaration, and a specified prefix name, and

13   installs the at least one interface definition language declaration in

14   a prefix naming context having the prefix naming context in the

15   interface repository;

16   a memory device that stores the interface repository; and

17   a processing unit that executes operations of the interface repository

18   loader, and further executes the interface repository loader to

19   create a data file identified as related to the specified interface

20   definition language file, and containing an identification of the

21   specified prefix naming context.

1   55. (New). A method of resolving name collisions by providing type

2 support for multiple type definitions, comprising the steps of:

3   defining in an interface repository a prefix naming context, the prefix

4   naming context being adapted to resolve names subordinate to the

5   repository naming context; and

6   storing the prefix naming context subordinate to the repository naming

7   context in the interface repository, the prefix naming context

8   forming an interface definition language root context for interface

9   definition objects subordinate to the prefix naming context.

1   56. (New) The method of claim 55, wherein each prefix naming context is

2 stored immediately subordinate to the repository naming context.

1   57. (New) The method of claim 55 further comprising the steps of:

2   specifying an interface definition language file containing at least one

3   interface definition language declaration;

4   specifying a prefix naming context; and

5   storing each interface definition language declaration in the specified

6   interface definition language file into the specified prefix naming

7   context.

1   58. (New) The method of claim 57, wherein the step of storing each

2 interface definition language declaration further comprises the steps of:

3   creating an interface definition object for the interface definition language

4   declaration;

5   storing the interface definition object in the specified prefix naming

6   context; and

7   providing the interface definition object with a fully scoped object name

8   including a prefix name from the prefix naming context in which

9   the interface definition object is stored.

1   59. (New) A method of resolving name collisions by providing type

2 support for multiple type definitions, comprising the steps of:

3   defining in an interface repository a prefix naming context, the prefix

4   naming context being adapted to resolve names subordinate to a

5   repository naming context;

6   storing the prefix naming context subordinate to the repository naming

7   context in the interface repository, the prefix naming context

8   forming an interface definition language root context for interface

9   definition objects subordinate to the prefix naming context;

10   specifying an interface definition language file containing at least one

11   interface definition language declaration;

12   specifying a prefix naming context;

13   storing each interface definition language declaration in the specified

14   interface definition language file into the specified prefix naming

15   context; and

16   creating a data file identified as related to the specified interface definition

17   language file, and containing an identification of the specified

18   prefix naming context.

1   60. (New) A method of resolving name collisions by providing type

2 support for multiple type definitions, comprising the step of:

3   providing an interface repository including:

4   a repository naming context; and

5   a prefix naming context subordinate to the repository naming

6   context, the prefix naming context serving as a root naming

7   context for at least one interface definition language

8   declaration, the prefix naming context being adapted to re-

9   solve names subordinate to the repository naming context.